Please amend the claims to read as indicated in the following list of claims:

1. [Currently amended] A method of identifying uncorrectable codewords in a Reed-Solomon decoder handling errors and erasures, the Reed-Solomon decoder including at least six hardware logic functions, the method comprising the steps of:

indicating an uncorrectable codeword when any one or more of the following conditions (a) to (f) is satisfied:

(a) no solution to key equation $\sigma(x)T(x) \equiv (x) \bmod x^{2T}$ is found by a first one of said at least six hardware logic functions;

(b) $\deg \sigma(x) \neq$ nerrors is determined by a second one of said at least six hardware logic functions;

(c) error and erasure locations coincide is determined by a third one of said at least six hardware logic functions;

(d) $\deg \omega(x) \geqq$ nerrors+nerasures is determined by a forth one of said at least six hardware logic functions;

(e) nerasures+2*nerrors>2T is determined by a fifth one of said at least six hardware logic functions and

(f) an error location has a zero correction magnitude is determined by a sixth one of said at least six hardware logic functions;

where nerrors and nerasures represent, respectively, a number of errors with reference to an error locator polynomial $\sigma(x)$ and a number of erasures with reference to an erasure locator polynomial $\Lambda(x)$, 2T is the strength of a Reed-Solomon code, $\omega(x)$ is an errata evaluator polynomial, and $T(x)$ is a modified syndrome polynomial.

2. [Original] The method of claim 1, comprising evaluating the condition (a) as a preliminary step, and then evaluating the conditions (b) to (f).

3. [Original] The method of claim 1, wherein the method comprises identifying a codeword as correctable if none of at least the conditions (a) to (f) are satisfied.

4. [Original] The method of claim 1, wherein the method comprises indicating an uncorrectable codeword in response to condition (g) deg $\Lambda(x) \neq$ nerasures.

5. [Original] The method of claim 1, wherein the method comprises receiving the error locator polynomial $\sigma(x)$, the erasure locator polynomial $\Lambda(x)$ and the errata evaluator polynomial $\omega(x)$; forming a set of error locations, and a set of erasure locations, and forming variables nerrors and nerasures representing the size of each set, respectively; and finding deg $\sigma(x)$, deg $\Lambda(x)$, and deg $\omega(x)$, as a degree of the error locator polynomial $\sigma(x)$, the erasure locator polynomial $\Lambda(x)$ and the errata evaluator polynomial $\omega(x)$, respectively.

6. [Original] A detector circuit arranged to identify an uncorrectable codeword, for use in a Reed-Solomon decoder handling errors and erasures, the circuit comprising:

a logic unit arranged to identify each condition:

(a) no solution to key equation $\sigma(x)T(x) \equiv (x) \bmod x^{2T}$;

(b) deg $\sigma(x) \neq$ nerrors;

(c) error and erasure locations coincide;

(d) $\deg\omega(x) \geqq$ nerrors+nerasures;

(e) nerasures+2*nerrors>2T; and

(f) an error location has a zero correction magnitude;

where nerrors and nerasures represents, respectively, a number of errors and erasures with reference to an error locator polynomial $\sigma(x)$ and an erasure locator polynomial $\Lambda(x)$, 2T is the strength of a Reed-Solomon code, $\omega(x)$ is an errata evaluator polynomial, and T(x) is a modified syndrome polynomial; and

an indicator unit arranged to indicate an uncorrectable codeword, responsive to the logic unit.

7. [Original] The circuit of claim 6, wherein the circuit comprises a counter arranged to count nerrors and nerasures as the size of a set of error locations derived from the error locator polynomial $\sigma(x)$, and a set of erasure locations derived from the erasure locator polynomial $\Lambda(x)$, respectively.

8. [Original] The circuit of claim 6, wherein the logic unit is arranged to identify an uncorrectable codeword in response to condition (g) $\deg\Lambda(x) \neq$ nerasures.

Claims 9-10. Canceled.

11. [New] A method of identifying uncorrectable Reed-Solomon codewords in a group Reed-Solomon codewords being passed to a Reed-Solomon decoder, the Reed-Solomon decoder including a polynomial generation unit, a polynomial

evaluation unit, an uncorrectable error detector and an error correction block, the method comprising:

generating polynomial data in the polynomial generation unit representing locations and magnitudes of errors and erasure in a Reed-Solomon codeword passed to the Reed-Solomon decoder,

performing a Chien search on said polynomial data in the polynomial evaluation unit and solving Forney's equations to determine correction locations and magnitudes to be applied to the Reed-Solomon codeword passed to the Reed-Solomon decoder, and

detecting, in the uncorrectable error detector, that the Reed-Solomon codeword passed to the Reed-Solomon decoder is an uncorrectable codeword when any one or more of the following conditions (a) to (f) is satisfied:

(a) no solution to key equation $\sigma(x)T(x) \equiv (x) \bmod x^{2T}$ is found by a first logic means;

(b) deg $\sigma(x) \neq$ nerrors is determined by a second logic means;

(c) error and erasure locations coincide is determined by a third logic means;

(d) deg$\omega(x) \geqq$ nerrors+nerasures is determined by a forth logic means;

(e) nerasures+2*nerrors>2T is determined by a fifth logic means and

(f) an error location has a zero correction magnitude is determined by a sixth logic means;

where nerrors and nerasures represent, respectively, a number of errors with reference to an error locator polynomial $\sigma(x)$ and a number of erasures with reference to an erasure locator polynomial $\Lambda(x)$, 2T is the strength of

a Reed-Solomon code, $\omega(x)$ is an errata evaluator polynomial, and T(x) is a modified syndrome polynomial; and

if the Reed-Solomon codeword passed to the Reed-Solomon decoder is not an uncorrectable codeword, then performing error correction on the Reed-Solomon codeword passed to the Reed-Solomon decoder using the correction locations and magnitudes determined by solving Forney's equations.